Land Price Forecasting

Zarikunte Kunal Dayanand (20315)

Abstract:

The report on land price forecasting presents a comprehensive analysis of the factors that influence the price of land in a particular area. The study involves the use of various machine learning algorithms to predict land prices based on the available data. The datasets used contain two different kinds of data frames one involving land and the other one including amenities. The report covers the entire process of data collection, pre-processing, and feature selection, followed by the application of different models to predict land prices. The results obtained from these models are then compared to identify the most accurate model. The report concludes with the identification of the most important features that contribute to land price prediction, providing valuable insights for real estate investors and analysts. Overall, this report aims to provide a practical approach to forecasting land prices that can be applied in real-world scenarios.

I. Introduction and Background:

Land price forecasting is a crucial task in the real estate industry. It involves predicting the market value of a piece of land, which is influenced by various factors such as location, neighbourhood amenities, and economic conditions. Accurate land price forecasting can aid in making informed investment decisions, assessing property taxes, and negotiating property sales.

In this report, we will discuss the use of Python in land price forecasting and present an example of a machine learning model that predicts land prices based on neighbourhood amenities and other factors. The model is trained on a dataset of land prices and amenities in a particular region and is evaluated using various performance metrics. The report will provide insights into the data analysis and modelling process, as well as recommendations for improving the accuracy of land price forecasting.

II. Methodology:

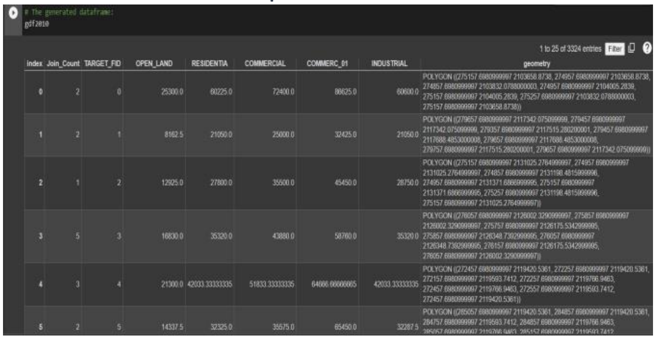
We have basically worked on the following principles:

1. Load the data- Loading the data containing land prices for different locations and dates into a geopandas dataframe.
2. Explore the data- Use geospatial visualization tools to explore the data, including heatmaps, scatter plots.
3. Data preparation- Preprocess the data by cleaning missing data, heading outliersand feature engineering to extract meaningful features from the data.
4. Feature selection: Select the most relevant features from the data to use in the forecasting model.
5. Training the model: Train a regression model on the selected features to predict land prices.
6. Evaluating the model: Evaluate the accuracy of the model using various metrics, such as mean squared error (MSE) and R-squared.
7. Making predictions: Use the trained model to make predictions on new data.

Step I: Loading the data and importing all the required libraries. Then going through the dataset, we used, it involves the data frames from 2006-2016 except that for 2010 that is for the 10 years. So firstly, we predicted the data for 2010 by two methods:

1. Averaging the data from 2009 and 2011.
2. Using Linear regression for the prediction.

And found out the results from averaging to be more useful so considering them. Out of that few of the entries are depicted as below:



```
# The generated dataframe:
gdf2010
```

1 to 25 of 3324 entries Filter

index	Join_Count	TARGET_FID	OPEN_LAND	RESIDENTIA	COMMERCIAL	COMMERC_01	INDUSTRIAL	geometry
0	2	0	25300.0	60225.0	72400.0	88625.0	60600.0	POLYGON ((275157.6980999997 2103658.8738, 274957.6980999997 2103658.8738, 274857.6980999997 2103832.0788000003, 274957.6980999997 2104005.2839, 275157.6980999997 2104005.2839, 275257.6980999997 2103832.0788000003, 275157.6980999997 2103658.8738))
1	2	1	8162.5	21050.0	25000.0	32425.0	21050.0	POLYGON ((279657.6980999997 2117342.075099999, 279457.6980999997 2117342.075099999, 279357.6980999997 2117515.280200001, 279457.6980999997 2117688.4853000008, 279657.6980999997 2117688.4853000008, 279757.6980999997 2117515.280200001, 279657.6980999997 2117342.075099999))
2	1	2	12925.0	27800.0	35500.0	45450.0	28750.0	POLYGON ((275157.6980999997 2131025.2764999997, 274957.6980999997 2131025.2764999997, 274857.6980999997 2131198.4815999996, 274957.6980999997 2131371.6866999995, 275157.6980999997 2131371.6866999995, 275257.6980999997 2131198.4815999996, 275157.6980999997 2131025.2764999997))
3	5	3	16830.0	35320.0	43880.0	58760.0	35320.0	POLYGON ((276057.6980999997 2126002.3290999997, 275857.6980999997 2126002.3290999997, 275757.6980999997 2126175.5342999995, 275857.6980999997 2126348.7392999995, 276057.6980999997 2126348.7392999995, 276157.6980999997 2126175.5342999995, 276057.6980999997 2126002.3290999997))
4	3	4	21300.0	42033.33333335	51833.33333335	64666.66666665	42033.33333335	POLYGON ((272457.6980999997 2119420.5361, 272257.6980999997 2119420.5361, 272157.6980999997 2119593.7412, 272257.6980999997 2119766.9463, 272457.6980999997 2119766.9463, 272557.6980999997 2119593.7412, 272457.6980999997 2119420.5361))
5	2	5	14337.5	32325.0	35675.0	65450.0	32287.5	POLYGON ((285057.6980999997 2119420.5361, 284857.6980999997 2119420.5361, 284757.6980999997 2119593.7412, 284857.6980999997 2119766.9463,

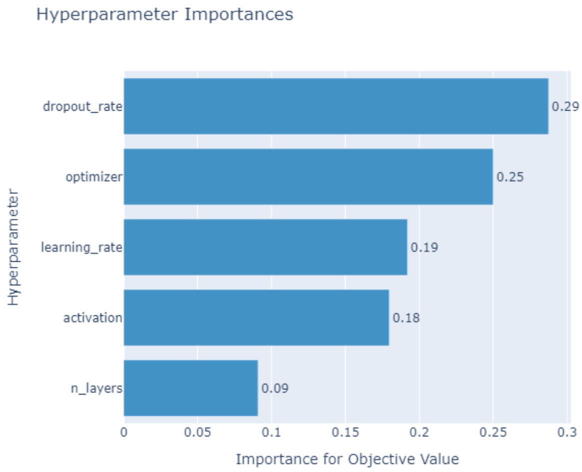
Step II: Inserting the dataframe to the list of each year data we used as dataset. We then used the same for constructing hexagonal heatmaps on the datasets as on price of Open land, Residential land, Commercial land, Commercial_01 land and Industrial land.
After all this we analysed the amenities dataset and made some distribution plots for the particular datas like using Facility ID and Count Store.

Step III: Now comes the Data processing part here the given code defines a function named datamaker that takes a property type as input and returns a DataFrame containing the corresponding price data for that property type for each year from 2006 to 2016. The function initializes an empty list named data and creates a new DataFrame named df_price from it. It then adds a column named 'TARGET_FID' to df_price with values from the 'TARGET_FID' column of the df_amenities DataFrame. The function then iterates through each DataFrame in the files list, extracts the price data for the specified property type, and adds it as a new column to df_price, with the year as the column name.

Step IV: Now we do Feature selection. For the problem we are using Numerical Feature Selection.
There are two popular feature selection techniques that can be used for numerical input data and a numerical target variable. They are:

1. Correlation Statistics.
2. Mutual Information Statistics.

Step V:
After selecting the data and important variables we split the data into train and test part to build deep learning model. We used three types of deep neural network for the forecasting part. Later the performance of the models were improved using the Optuna python library that tuned the hyperparameters. Hyper parameters that were tuned included activation function, number of layers, dropout rate and learning rate. The importance of each hyper parameter is visualized as follows



III. Results and Discussion:

1. Hexagonal price heatmaps obtained are as follows:

a. Open land

b. Residential land

c. Commercial land

d. Commercial_01 land

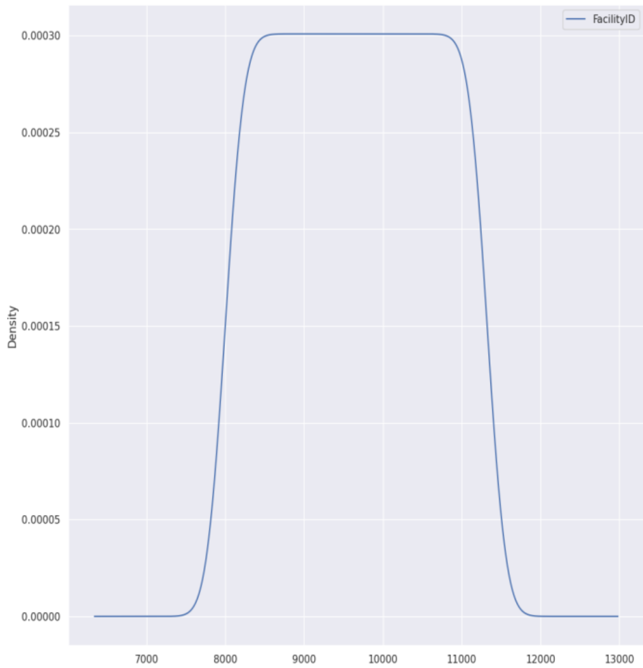
e. Industrial land

2. Amenities data analysis

Distribution plots:

a. Facility ID

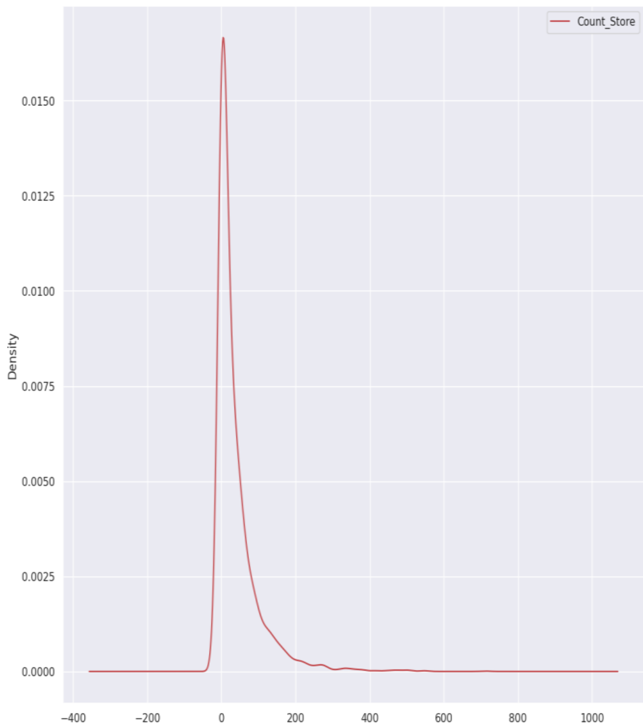
Almost uniform distribution



f. Complete

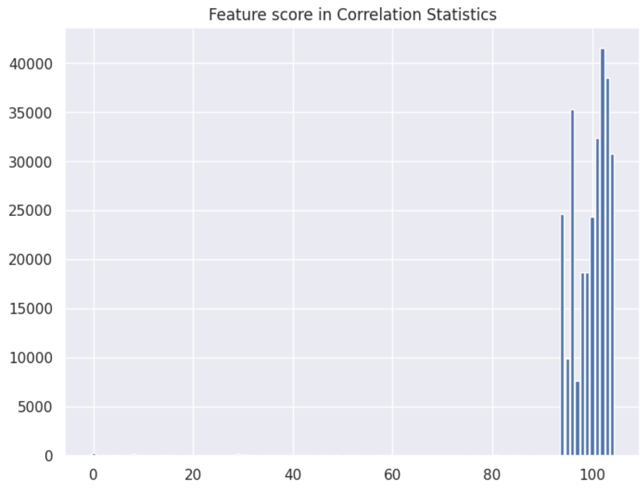
b. Count Store

Skewed distribution

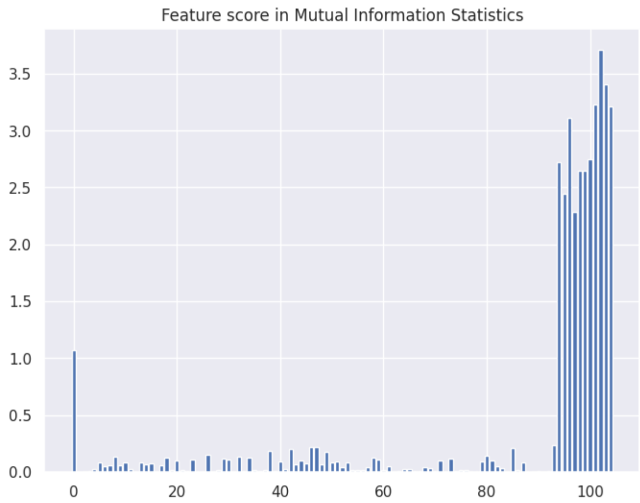


3. Feature Selection

a. Correlation Statistics



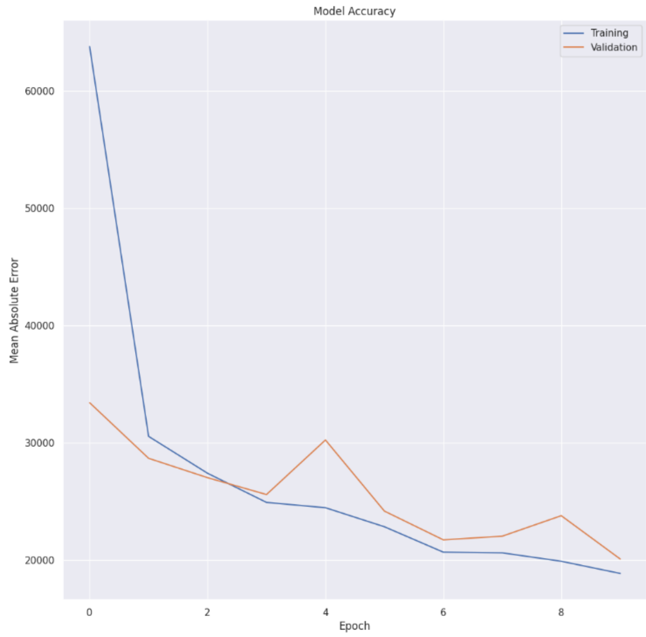
b. Mutual Information Statistics



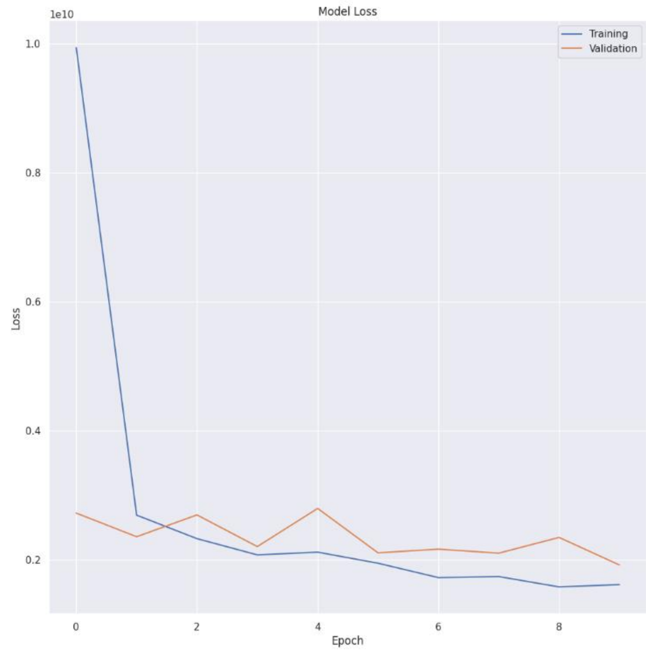
4. We have used three types of deep neural network for the forecasting part and the results obtained are as follows:

a. 1st NN

i.a) Model Accuracy

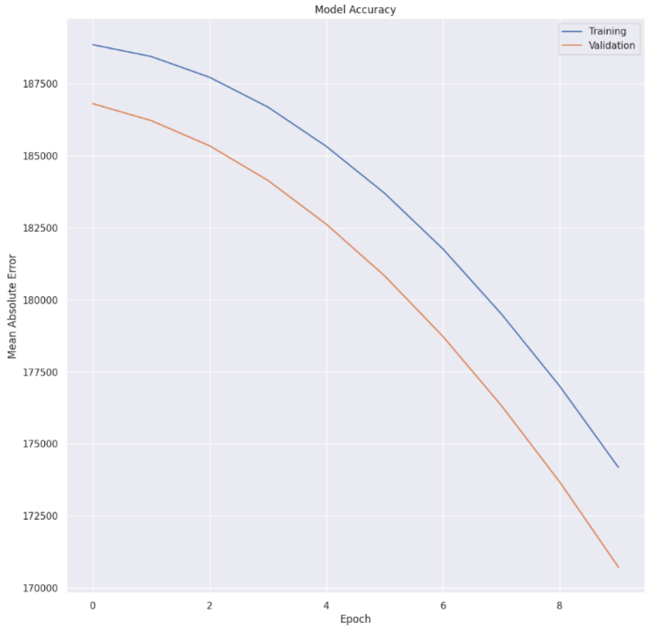


ii.a) Model Loss

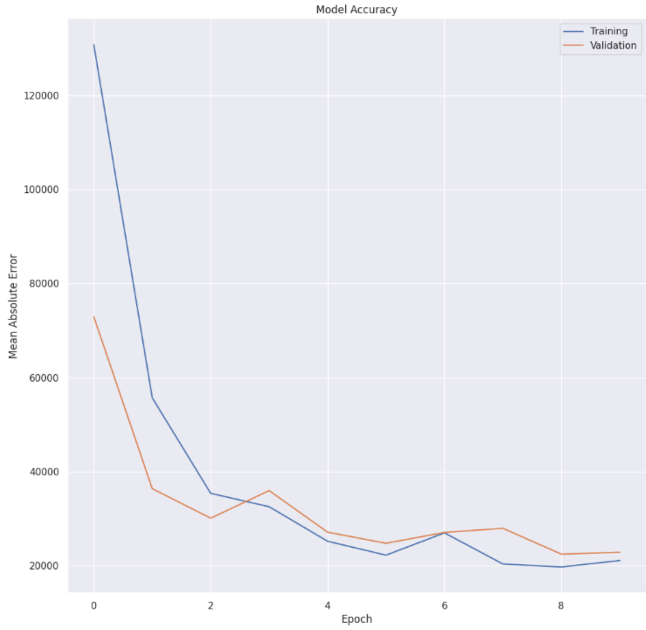


b. 2nd NN

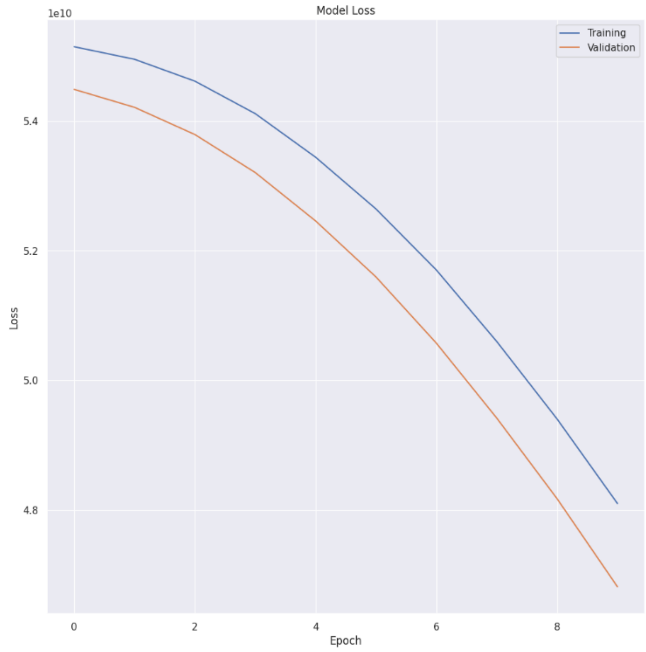
i.b) Model Accuracy



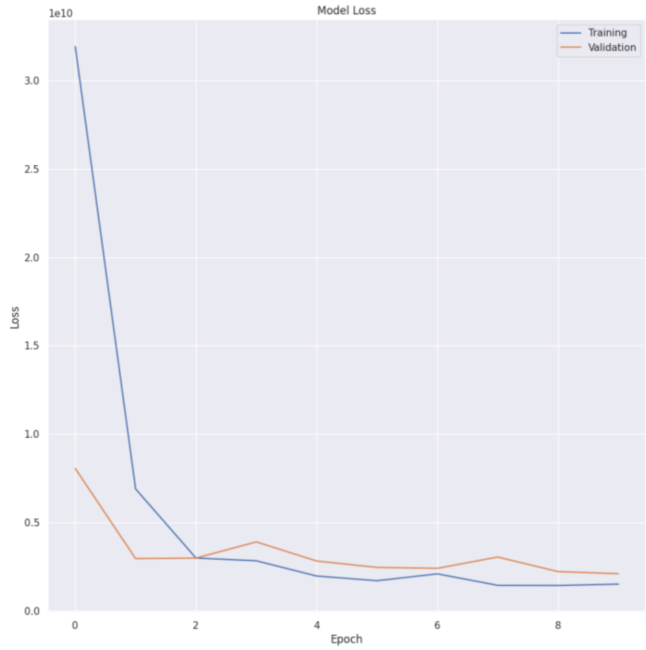
i.c) Model Accuracy



ii.b) Model Loss

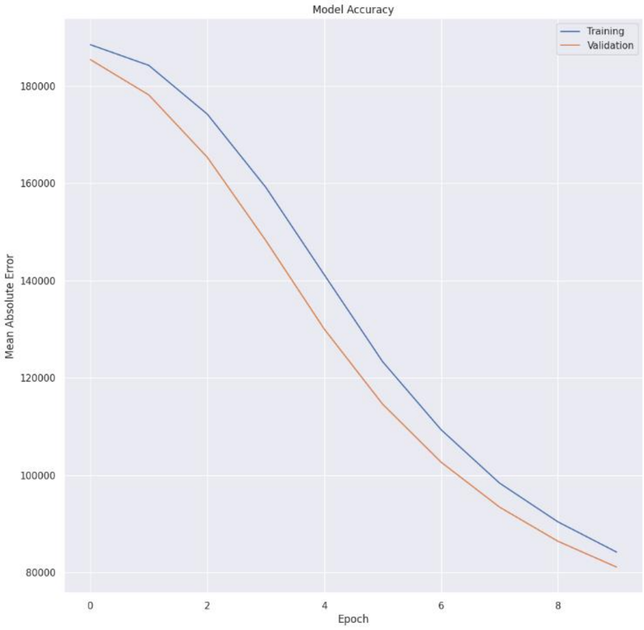


ii.c) Model Loss

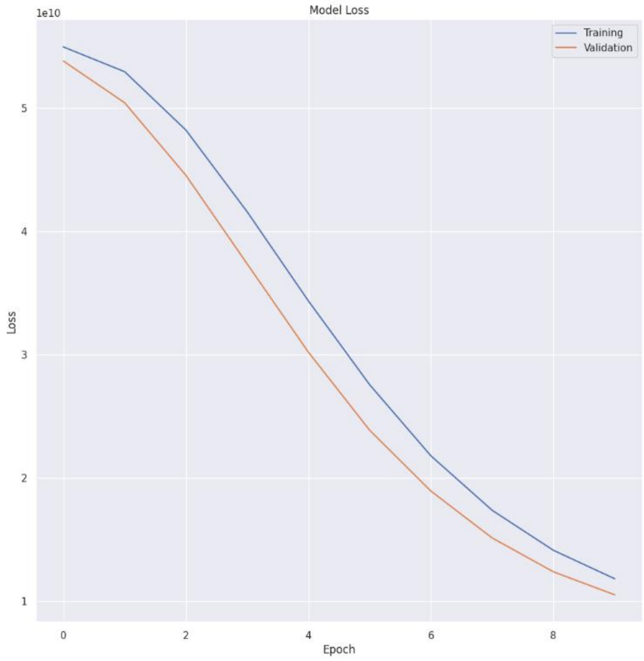


c. 3rd NN

a. Model Accuracy



b. Model Loss



IV. Conclusion:

The land price forecasting model developed in this project can be useful for real estate agents, property developers, and investors who want to make informed decisions about buying or selling land. By accurately predicting the future prices of land, the model can help these stakeholders to optimize their investment strategies and maximize their profits.

V. References:

1. @misc{chollet2015keras,title={Keras},author={Chollet, Fran\c{c}ois and others}, year={2015}, howpublished={\url{https://keras.io}},
2. Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo,Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis,Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.
3. @software{reback2020pandas,autho={The pandas development team,} title={pandas-dev/pandas: Pandas}
4. Bradski, G., 2000. The OpenCV Library. Dr. Dobb's Journal of Software Tools.
5. Harris, C.R., Millman, K.J., van der Walt, S.J. et al. Array programming with NumPy. Natur